



(11)

EP 3 550 802 A1

(12)

EUROPEAN PATENT APPLICATION
published in accordance with Art. 153(4) EPC

(43) Date of publication:
09.10.2019 Bulletin 2019/41

(51) Int Cl.:
H04L 29/08 ^(2006.01)

(21) Application number: **17916475.1**

(86) International application number:
PCT/CN2017/095551

(22) Date of filing: **02.08.2017**

(87) International publication number:
WO 2019/006808 (10.01.2019 Gazette 2019/02)

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA ME
Designated Validation States:
MA MD

(30) Priority: **07.07.2017 CN 201710550234**

(71) Applicant: **Wangsu Science & Technology Co., Ltd.**
Shanghai 200030 (CN)

(72) Inventors:
• **YU, Shengshui (CN)**
• **LIN, Zhangkun (CN)**

(74) Representative: **Hanna Moore + Curley**
Garryard House
25/26 Earlsfort Terrace
Dublin 2, D02 PX51 (IE)

(54) **METHOD AND DEVICE FOR PROCESSING PERSISTENT CONNECTION ESTABLISHMENT REQUEST**

(57) The present disclosure provides a method and a device for processing a persistent connection establishment request, which belongs to the technical field of network communication. The method includes: receiving the persistent connection establishment request sent by a client; determining a first processing process with a lowest number of current persistent connections, based

on a locally stored number of current persistent connections of each processing process; and using the first processing process to process the persistent connection establishment request. By using the present disclosure, the utilization rate of the processing resources by the network service server may be improved.

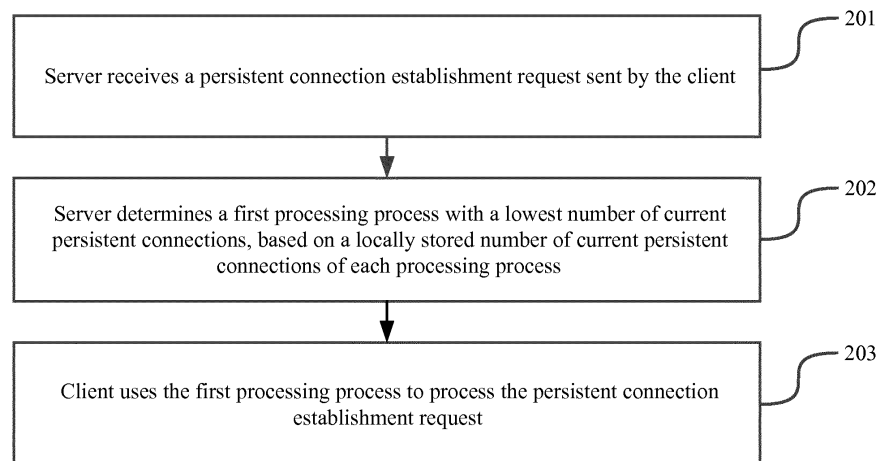


FIG. 2

EP 3 550 802 A1

DescriptionFIELD OF THE DISCLOSURE

[0001] The present disclosure relates to the field of network communication technologies and, more particularly, relates to a method and a device for processing a persistent connection establishment request.

BACKGROUND

[0002] As the network technologies develop rapidly, the network services are becoming more diversified and complicated. Currently, the network services based on the TCP (Transmission Control Protocol) connection may include two connection modes: persistent connection and multiple connections.

[0003] The persistent connections are usually applied to point-to-point communication showing frequent operations. The user terminal may establish a TCP connection with the client provided by the service supplier, and meanwhile, the client may establish a TCP connection with a server.

[0004] After the aforementioned TCP connection is established and before it is released, it can be ensured that the TCP connection stays connected. As such, when data interaction demand exists between the user terminal and the server, data interaction with the server may be performed through the client, and in each data interaction, data packet(s) may be sent based on the established TCP connection without the TCP connection being re-established. Accordingly, processing resources consumed by establishment of the TCP connection may be effectively saved.

[0005] During the process of implementing the present disclosure, the inventors discover at least following issues in existing technologies.

[0006] After the client establishes the persistent connection with the server, the server often maintain persistent connections with several clients, respectively, through a plurality of processes. Due to the randomness of the services initiated by the user terminal, the situation in which a specific processing process of the server is overloaded while other processing processes are in an idle state may occur. Thus, the utilization rate of the processing resources by the server is relatively low.

BRIEF SUMMARY OF THE DISCLOSURE

[0007] To solve issues in existing technologies, embodiments of the present disclosure provide a method and a device for processing a persistent connection establishment request. The technical solutions are as follows.

[0008] In a first aspect, a method for processing a persistent connection establishment request is provided, the method includes:

receiving a persistent connection establishment request sent by a client;
based on the locally stored number of current persistent connections of each processing process, determining a first processing process with the lowest number of current persistent connections;
using the first processing process to process the persistent connection establishment request.

[0009] Optionally, receiving the persistent connection establishment request sent by the client comprises: determining a second processing process based on a preset receiving order, and receiving the persistent connection establishment request sent by the client through the second processing process.

[0010] Optionally, the method further includes:

through the second processing process, using SOCKET to forward the persistent connection establishment request to a managing process; and based on the locally stored number of current persistent connections of each processing process, determining the first processing process with the lowest number of current persistent connections includes: based on the locally stored number of current persistent connections of each processing process, determining the first processing process with the lowest number of current persistent connections through the managing process.

[0011] Optionally, after receiving the persistent connection establishment request sent by the client, the method further comprises:

if the persistent connection establishment request carries identification information of a third processing process, using the third processing process to process the persistent connection establishment request; and

based on the locally stored number of current persistent connections of each processing process, determining the first processing process with the lowest number of current persistent connections includes: if the persistent connection establishment request does not carry identification information of any processing process, based on the locally stored number of current persistent connections of each processing process, determining the first processing process with the lowest number of current persistent connections.

[0012] Optionally, based on the locally stored number of current persistent connections of each processing process, determining the first processing process with the lowest number of current persistent connections comprises:

based on the locally stored number of current per-

sistent connections of each processing process, determining a preset number of processing processes with the lowest number of current persistent connections;

acquiring identification information of the client carried in the persistent connection establishment request, and determining the first processing process with the lowest number of current persistent connections established with the client, among the preset number of the processing processes.

[0013] In a second aspect, a device for processing a persistent connection establishment request is provided, the device includes:

a receiving module, for receiving a persistent connection establishment request sent by a client;
a determination module for, based on the locally stored number of current persistent connections of each processing process, determining a first processing process with a lowest number of current persistent connections;
a processing module, for using the first processing process to process the persistent connection establishment request.

[0014] Optionally, the receiving module is specifically configured to:

determine a second processing process based on a preset receiving order, and receive the persistent connection establishment request sent by the client through the second processing process.

[0015] Optionally, the device further includes:

a forwarding module for, through the second processing process, using SOCKET to forward the persistent connection establishment request to a managing process; and
the determination module is specifically configured to:
based on the locally stored number of current persistent connections of each processing process, determining a first processing process with a lowest number of current persistent connections through the managing process.

[0016] Optionally, the processing module is further configured to:

after receiving the persistent connection establishment request sent by the client, if the persistent connection establishment request carries identification information of a third processing process, use the third processing process to process the persistent connection establishment request; and
the determination module is specifically configured to:
if the persistent connection establishment request

carries no identification information of any processing process, based on the locally stored number of current persistent connections of each processing process, determine the first processing process with the lowest number of current persistent connections.

[0017] Optionally, the determination module is specifically configured to:

based on the locally stored number of current persistent connections of each processing process, determine a preset number of processing processes with the lowest number of current persistent connections;

acquire identification information of the client carried in the persistent connection establishment request, and determine the first processing process with the lowest number of current persistent connections established with the client, among the preset number of the processing processes.

[0018] In the third aspect, a server is provided. The server includes a processor and a memory. The memory stores at least one instruction, at least one section of programs, a code set, or an instruction set. The at least one instruction, at least one section of programs, a code set or an instruction set is loaded by the processor and is executed to implement the method for processing a persistent connection establishment request, as described in the first aspect.

[0019] In the fourth aspect, a computer-readable storage medium is provided, and the storage medium stores at least one instruction, at least one section of programs, a code set or an instruction set. The at least one instruction, at least one section of programs, a code set or an instruction set is loaded by the processor and is executed to implement the method for processing a persistent connection establishment request, as described in the first aspect.

[0020] Beneficial effects brought about by the technical solutions provided by embodiments of the present disclosure are as follows.

[0021] In embodiments of the present disclosure, the server receives a persistent connection establishment request sent by the client, and based on the locally stored number of current persistent connections of each processing process, determines a first processing process with a lowest number of current persistent connection. Further, the first processing process processes the persistent connection establishment request. As such, when persistent connection needs to be established with the client, the server may, from all the processing processes, select the processing process that has the lowest load for processing, thereby enabling the loads of the plurality of processing processes to be balanced and improving the utilization rate of the processing resources by the server.

BRIEF DESCRIPTION OF THE DRAWINGS

[0022] To more clearly illustrate technical solutions in embodiments of the present disclosure, accompanying drawings to be used in the following embodiments will be introduced briefly. Obviously, the accompanying drawings described hereinafter are merely some embodiments of the present disclosure, and for those ordinarily skilled in the relevant art, other drawings may be derived from such drawings without contributing creative labor.

FIG. 1 illustrates a schematic view of a scenario in which a persistent connection establishment request is established according to embodiments of the present disclosure;

FIG. 2 illustrates a flow chart of a method for processing a persistent connection establishment request according to embodiments of the present disclosure;

FIG. 3 illustrates a structural schematic view of a system for processing a persistent connection establishment request according to embodiments of the present disclosure;

FIG. 4 illustrates a schematic view showing steps of processing a persistent connection establishment request according to embodiments of the present disclosure;

FIG. 5 illustrates a structural schematic view of a device for processing a persistent connection establishment request according to embodiments of the present disclosure;

FIG. 6 illustrates a structural schematic view of a device for processing a persistent connection establishment request according to embodiments of the present disclosure;

FIG. 7 illustrates a structural schematic view of a server according to embodiments of the present disclosure.

DETAILED DESCRIPTION

[0023] To more clearly describe the objectives, technical solutions and advantages of the present disclosure, implementations of the present disclosure is further illustrated in details with reference to the accompanying drawings in conjunction with embodiments.

[0024] Embodiments of the present disclosure provides a method for processing a persistent connection establishment request, and the execution body of the method may be a server. The server may be any server that provides network service, and the network service may be cloud broadband service, and cloud line service, etc. A plurality of processing processes responsible for

network services may run on the server, and each processing process may support at least one persistent connection with the client. The at least one client herein may be client(s) provided by the service supplier, and may be distributed over different regions to assist execution of network services initiated by user terminals in each region.

[0025] The specific scenario may refer to FIG. 1, where the user terminal may establish a connection with the aforementioned client, and perform data interaction with the client through the persistent connection between the client and the server.

[0026] The aforementioned server may include a processor, a memory and a transceiver. The processor may be configured to process a persistent connection establishment request in the following procedures. The memory may be configured to store data needed by the following processing procedure and the data generated thereof. The transceiver may be configured to receive and send related data in the following processing procedure, such as receive the information sent by the client, and return data to the client, etc.

[0027] With reference to the specific implementations described hereinafter, the processing procedure shown in FIG. 2 is illustrated in details, and the procedure is as follows.

[0028] At Step 201, the server receives a persistent connection establishment request sent by the client.

[0029] In the implementation, the service supplier of the network service may arrange clients in each region within the service coverage range to assist the user terminal in performing network service.

[0030] On one hand, a plurality of processing processes may run simultaneously on the client, and the processing processes may be primarily responsible for establishment/release of the persistent connection with the server and the sending of the persistent data. Each processing process of the client may establish a persistent connection with the server through the TCP protocol, and between each processing process of the client and the server, the establishment and release of the persistent connection are individual to each other, without mutual disturbance.

[0031] After the client enters a working status, the main process is started. Further, based on the configuration file, the managing process and the processing process are created by the main process, and the processing process listens to the service port to receive the network service data sent by the user terminal. Further, the client may forward the network service data sent by the user terminal to the server by establishing the persistent connection with the server. Here, the client may detect the loading condition of the current persistent connection, and if the load of the persistent connection reaches a threshold, the number of the persistent connections may increase dynamically, namely, a persistent connection establishment request may be sent to the server. Similarly, if during the detection process, the client discovers

that the number of idle persistent connections is greater than a preset threshold, a part of the persistent connections may be released dynamically. The managing process of the client may be used to manage the processing processes, such as fault detection, and statistics of loading, etc.

[0032] On the other hand, a plurality of processing processes may run simultaneously on the server. More specifically, when the server operates, a main process may be started, and based on the configuration file, a managing process and processing processes may be created by the main process. Further, connections between the managing process and each processing process may be established, and the load information of each processing process may be recorded through the managing process. When the client detects that a new persistent connection needs to be established with the server, and after the client sends a persistent connection establishment request to the server, the server may receive the corresponding persistent connection establishment request.

[0033] Optionally, the server may receive the persistent connection establishment request through a plurality of processing processes in turn. Correspondingly, processing of Step 201 may be as follows: determining a second processing process based on a preset receiving order, and receiving the persistent connection establishment request sent by the client through the second processing process.

[0034] In implementation, the server may be configured with a uniform service port configured to receive a persistent connection establishment request. The plurality of processing processes may all listen to the service port, and when the persistent connection establishment request arrives, the server may awake a processing process (i.e., the second processing process) based on the preset receiving order, and further receives the persistent connection establishment request sent by the client through the second processing process. Here, the preset receiving order may be the arrangement order of the reference number of each processing process, or may be the size order of the number of the persistent connections already established by each processing process, or may even be other orders configured manually by technicians, etc. It is worth mentioning that, different from using a single listening process to listen to the service port, a plurality of processing processes may be selected herein to listen to the service port at the same time. Thus, the situations where the number of persistent connection establishment requests that undergo parallel processing by the server is relatively low due to the insufficient upper limit of the load of the listening process may be effectively avoided.

[0035] At Step 202, based on the locally stored number of current persistent connections of each processing process, the server determines a first processing process with a lowest number of current persistent connections.

[0036] In implementation, the managing process of the

server may record load information of each processing process, and the load information may at least include the number of current persistent connections of each processing process. When one or more persistent connections are added to or deleted from a certain processing process, the managing process may modify the number of the persistent connections of the recorded processing process adaptively. As such, after the server receives the persistent connection establishment request sent by the client, the locally stored number of the persistent connections of each processing process may be first acquired, and the processing process having the lowest load may be selected. That is, the first processing process with the minimum persistent connections is determined.

[0037] Optionally, after the server receives the persistent connection establishment request through the second processing process, the persistent connection establishment request may be forwarded to the managing process, and the specific procedure may include: through the second processing process, using SOCKET to forward the persistent connection establishment request to the managing process. Correspondingly, the procedure of Step 202 may be as follows: based on the locally stored number of current persistent connections of each processing process, determining a first processing process with a lowest number of current persistent connections through the managing process.

[0038] In implementation, the server may enable a plurality of processing processes to listen to a service port by configuring the "SO_REUSEPORT" property of the SOCKET. Further, the server may implement communication between the processing processes and the managing process through the SOCKET, thereby achieving the objective of sharing information between processes. As such, in existing technologies, when each processing process shares the information, the loss in the system performance of the server caused by the usage of lock mechanism may be reduced. Based on the aforementioned procedures, the server determines the second processing process, and after using the second processing process to receive the persistent connection establishment request, the server may use the SOCKET to forward the persistent connection establishment request to the managing process of the server through the second processing process.

[0039] Further, the managing process of the server may receive the persistent connection establishment request forwarded by the second processing process, and thus may acquire the load information (i.e., the number of current persistent connections) of each processing process stored locally. Further, based on the acquired load information, the first processing process having the lowest number of persistent connections may be determined.

[0040] Optionally, the client may designate the processing process for processing the persistent connection establishment request, and the corresponding

process may be as follows: if the persistent connection establishment request carries identification information of a third processing process, using the third processing process to process the persistent connection establishment request. Further, the processing of Step 202 may be as follows: if the persistent connection establishment request does not carry any identification information of any processing process, based on the locally stored number of current persistent connections of each processing process, determining the first processing process with the lowest number of current persistent connections.

[0041] In implementations, when starting the persistent connection establishment request, the client may designate a processing process for performing processing. For example, for network services involving a relatively large data amount, such as download of movies and of installation packages of games, during the initial stage of data interaction, the service data may be transmitted together with service data of other network services in the same persistent connection. During transmission, if the client finds the data amount of the network service is relatively large and needs to individually occupy a persistent connection, the client may re-establish a persistent connection with the original processing process of the server. Thus, the identification information of the original processing process may be directly written into the persistent connection establishment request.

[0042] As such, the persistent connection establishment request received by the server may carry the identification information of a certain processing process. Further, the server may determine a corresponding processing process (i.e., a third processing process) through the identification information. Further, the third processing process may be applied to process the persistent connection establishment request. It can be understood that, if after receiving the persistent connection establishment request transmitted by the client, the server finds the request includes no identification information of any processing process, processing of Step 202 may be executed. That is, the first processing process having the lowest number of current persistent connections may be directly determined based on the locally stored number of current persistent connections of each processing process.

[0043] It should be noted that, based on the processing of using the second processing process to receive the persistent connection establishment request, after the second processing process is received, whether the persistent connection establishment request carries identification information of the second processing process may be first determined. If yes, the second processing process may be used to directly process the persistent connection establishment request. If the persistent connection establishment request carries identification information of other processing processes or does not carry any identification information of any processing process, the server may use the SOCKET or the second process-

ing process to forward the persistent connection establishment request to the managing process.

[0044] Optionally, when selecting a processing process to process the persistent connection establishment request, a plurality of persistent connection of the same client may be allocated as evenly as possible between a plurality of processing processes. Correspondingly, the processing of Step 202 may be as follows: based on the current persistent connections of each locally stored processing process, determining a preset number of processing processes with the lowest number of the current persistent connections; acquiring the identification information of the client carried in the persistent connection establishment request, and during the preset number of processing processes, determine the first processing process that has the lowest number of persistent connections established with the client.

[0045] In implementation, the managing process of the server may store a table of load information corresponding to each processing process, and each table of load information may include a plurality of records. Each record may correspond to one client and store the number of the persistent connections already established with the client in the current processing process. Thus, after the server receives a persistent connection establishment request sent by the client, the number of the persistent connections of each locally stored processing process may be acquired, and the plurality of processing processes having the lowest current load may be selected. That is, a preset number of processing processes having the lowest number of persistent connection established with the client may be determined. Further, among the preset number of processing processes, the first processing process with the lowest number of persistent connections established with the client may be determined.

[0046] It should be understood that, because the load condition of each client is different, the load amount of a persistent connection established by each client with the server is different. As such, even when the number of persistent connections established on two processing processes are the same, a relatively large difference may exist between the load amounts of the two processing processes. Thus, when selecting a processing process to process the request from the client to establish a connection, it should be ensured that the persistent connection establishment requests from the same client need to be as evenly distributed to a plurality of processing processes as possible for processing.

[0047] At Step 203, the client uses the first processing process to process the persistent connection establishment request.

[0048] In implementation, after determining the first processing process with the minimum number of current persistent connections, the client may use the first processing process to process the aforementioned persistent connection establishment request, namely, to establish a new persistent connection with the client

through the first processing process. Further, the managing process may update the load information of the first processing process. That is, the number of the current persistent connections of the first processing process may be increased by one.

[0049] It should be noted that, the disclosed client and server are models using multiplexed processes and multiple I/Os (Input/Output). As shown in FIG. 3, each processing process is individual to each other, which does not affect the stability of the main process and may fully utilize the system resources such as the multi-core CPU (central processing units).

[0050] For ease of understanding, FIG. 4 illustrate the specific processing procedure of the disclosed server, where the main process may create the managing process and processing processes after being started. All processing processes listen to the service port, and the second processing process is selected to receive the persistent connection establishment request based on a certain order. Further, whether the persistent connection establishment request needs to be processed by the second processing process is determined. If yes, the request is processed by the second processing process, and if not, the persistent connection establishment request is forwarded to the managing process, and the managing process determines whether or not to designate a processing process. If yes, the request is processed by the designated third processing process; and if not, the first processing process is selected for processing the request according to a preset load balancing algorithm.

[0051] In embodiments of the present disclosure, the server receives the persistent connection establishment request sent by the client, and based on the locally stored number of current persistent connections of each processing process, the server determines the first processing process with the minimum number of current persistent connections and uses the first processing process to process the persistent connection establishment request. As such, when the server needs to establish the persistent connection with the client, the server may, from all the processing processes, select the processing process with the lowest load for processing. Accordingly, the load of the plurality of processing processes become balanced, and the utilization rate of server in processing the resources is improved.

[0052] Based on the same technical idea, embodiments of the present disclosure further provide a device for processing a persistent connection establishment request. As shown in FIG. 5, the device include followings.

[0053] A receiving module 501 is configured to receive a persistent connection establishment request sent by a client.

[0054] A determination module 502 is configured to, based on the locally stored number of current persistent connections of each processing process, determine a first processing process with a lowest number of current persistent connections.

[0055] A processing module 503 is configured to use

the first processing process to process the persistent connection establishment request.

[0056] Optionally, the receiving module 501 is specifically configured to:

5 determine a second processing process based on a preset receiving order, and receive the persistent connection establishment request sent by the client through the second processing process.

[0057] Optionally, as shown in FIG. 6, the device further includes:

10 a forwarding module configured to, through the second processing process, use SOCKET to forward the persistent connection establishment request to a managing process.

[0058] The determining module 502 is specifically configured to:

based on the locally stored number of current persistent connections of each processing process, determine a first processing process with a lowest number of current persistent connections through the managing process.

[0059] Optionally, the processing module 503 is further configured to:

after receiving the persistent connection establishment request sent by the client, if the persistent connection establishment request carries identification information of a third processing process, use the third processing process to process the persistent connection establishment request.

[0060] The determining module 502 is specifically configured to:

30 if the persistent connection establishment request carries no identification information of any processing process, based on the locally stored number of current persistent connections of each processing process, determine the first processing process with the lowest number of current persistent connections

[0061] Optionally, the determining module 502 is specifically configured to:

40 based on the locally stored number of current persistent connections of each processing process, determine a preset number of processing processes with the lowest number of current persistent connections;

45 acquire identification information of the client carried in the persistent connection establishment request, and determine the first processing process with the lowest number of current persistent connections established with the client, among the preset number of the processing processes.

[0062] In embodiments of the present disclosure, the server receives the persistent connection establishment request sent by the client, and based on the locally stored number of current persistent connections of each processing process, the server determines the first processing process with the minimum number of current persistent connections and uses the first processing

process to process the persistent connection establishment request. As such, when the server needs to establish the persistent connection with the client, the server may select, from all the processing processes, the processing process with the lowest load for processing. Accordingly, the loads of the plurality of processing processes become balanced, and the utilization rate of server in processing the resources is improved.

[0063] It should be noted that: when the device for processing the persistent connection establishment request provided by the aforementioned embodiments processes the persistent connection establishment request, the division of the aforementioned functional modules are merely for illustrative purposes. In practical applications, the aforementioned functions may be implemented by different functional modules based on demand. That is, the internal structure of the device may be divided into different functional modules to implement all or partial of the above-described functions. Further, the device for processing the persistent connection establishment request provided by the aforementioned embodiments is based on the same idea behind the method embodiment for processing the persistent connection establishment request belong to the same idea, whose specific implementation process may refer to the method embodiments. Repeated descriptions are not provided herein.

[0064] FIG. 7 is a structural schematic view of a server provided by embodiments of the present disclosure. The server 700 may vary largely due to the variance of configurations and performances, which may include one or more CPUs 722 (e.g., one or more processors), a memory 732, one or more storage application programs 742 or a storage medium 730 (e.g., one or more mass storage devices) of data 744. The memory 732 and the storage medium 730 may be temporary or persistent storage media. Programs stored in the storage medium 730 may include one or more modules (not labeled in the figure), and each module may include a series of instructional operations in the server. Further, the CPU 722 may be configured to communicate with the storage medium 730, and at the server 700, a series of instructional operations in the storage medium may be executed.

[0065] The server 700 may further include one or more power sources 726, one or more wired or wireless network interfaces 750, one or more I/O interfaces 758, one or more keyboards 756, and/or one or more operating systems 741, such as Windows Server™, Mac OS X™, Unix™, Linux™, and FreeBSD™.

[0066] The server 700 may include a memory, and one or more programs. The one or more programs may be stored in the memory, and may be configured to enable one or more processors to execute one or more programs including instructions for processing the aforementioned persistent connection establishment request.

[0067] Those ordinarily skilled in the relevant art may understand that all or partial steps of the aforementioned embodiments may be implemented by hardware, or may

be implemented by using programs to instruct related hardware. The programs may be stored in the computer-readable storage medium, and the aforementioned medium may be a read-only memory, a magnetic disc, or an optic disc.

[0068] The foregoing is only preferred embodiment of the present disclosure, and is not intended to limit the present disclosure. Without departing from the spirit and principles of the present disclosure, any modifications, equivalent replacements and improvements shall all fall within the protection scope of the present disclosure.

Claims

1. A method for processing a persistent connection establishment request, comprising:

receiving, by a server, the persistent connection establishment request sent by a client; and determining, by a server, a first processing process with a lowest number of current persistent connections, based on a locally stored number of current persistent connections of each processing process, wherein the first processing process is used to process the persistent connection establishment request.

2. The method according to claim 1, wherein receiving the persistent connection establishment request sent by the client comprises:

determining a second processing process based on a preset receiving order, and receiving the persistent connection establishment request sent by the client through the second processing process.

3. The method according to claim 2, further comprising:

using a SOCKET to forward the persistent connection establishment request to a managing process, by the second processing process; and the determining of the first processing process with the lowest number of current persistent connections, based on the locally stored number of current persistent connections of each processing process, includes:

determining the first processing process with the lowest number of current persistent connections, by the managing process, based on the locally stored number of current persistent connections of each processing process.

4. The method according to claim 1, after the receiving of the persistent connection establishment request sent by the client, further comprising:

- if the persistent connection establishment request carries identification information of a third processing process, using the third processing process to process the persistent connection establishment request; and
the determining the first processing process with the lowest number of current persistent connections, based on the locally stored number of current persistent connections of each processing process, includes:
if the persistent connection establishment request does not carry any identification information of any processing process, determining the first processing process with the lowest number of current persistent connections, based on the locally stored number of current persistent connections of each processing process.
5. The method according to claim 1-4, wherein the determining of the first processing process with the lowest number of current persistent connections, based on the locally stored number of current persistent connections of each processing process, comprises:
- determining a preset number of processing processes with the lowest number of current persistent connections, based on the locally stored number of current persistent connections of each processing process; and
acquiring identification information of the client carried in the persistent connection establishment request, and determining the first processing process with the lowest number of persistent connections established with the client, among the preset number of the processing processes.
6. A server for processing a persistent connection establishment request, comprising:
- a memory, configured to store program instructions for performing a method for processing the persistent connection establishment request; and
a processor, coupled with the memory and, when executing the program instructions, configured to:
- receive a persistent connection establishment request sent by a client; and
determine a first processing process with a lowest number of current persistent connections, based on a locally stored number of current persistent connections of each processing process, wherein the first processing process is used to process the persistent connection establishment request.
7. The server according to claim [[1]]6, wherein the processor is further configured to:
determine a second processing process based on a preset receiving order, and receive the persistent connection establishment request sent by the client through the second processing process.
8. The server according to claim 7, wherein the processor is further configured to:
forward the persistent connection establishment request to a managing process, by the second processing process, using a SOCKET; and
determine the first processing process with the lowest number of current persistent connections through the managing process, based on the locally stored number of current persistent connections of each processing process.
9. The server according to claim 6, wherein the processor is further configured to:
after receiving the persistent connection establishment request sent by the client, if the persistent connection establishment request carries identification information of a third processing process, use the third processing process to process the persistent connection establishment request; and
if the persistent connection establishment request does not carry any identification information of any processing process, determine the first processing process with the lowest number of current persistent connections, based on the locally stored number of current persistent connections of each processing process.
10. The server according to claim 6-9, wherein the processor is further configured to:
determine a preset number of processing processes with the lowest number of current persistent connections, based on the locally stored number of current persistent connections of each processing process; and
acquire identification information of the client carried in the persistent connection establishment request, and determine the first processing process with the lowest number of persistent connections established with the client, among the preset number of the processing processes.
11. A server, **characterized in that**, the server includes a processor and a memory, the memory stores at least one instruction, at least one section of programs, a code set, or an instruction set, and the at least one instruction, the at least one section of programs, the code set or the instruction set is loaded

by the processor and is executed to implement the method for processing the persistent connection establishment request according to any one of claims 1-5.

5

12. A computer-readable storage medium, **characterized in that**, the storage medium stores at least one instruction, at least one section of programs, a code set or an instruction set, and the at least one instruction, the at least one section of programs, the code set or the instruction set is loaded by the processor and is executed to implement the method for processing a persistent connection establishment request according to any one of claims 1-5.

10

15

20

25

30

35

40

45

50

55

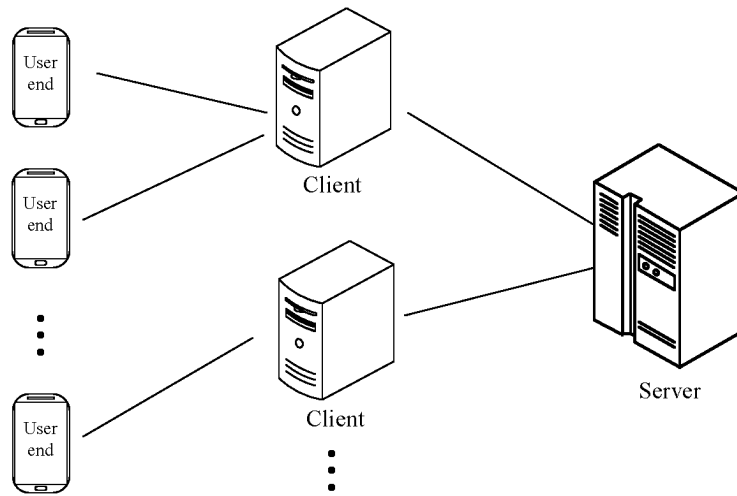


FIG. 1

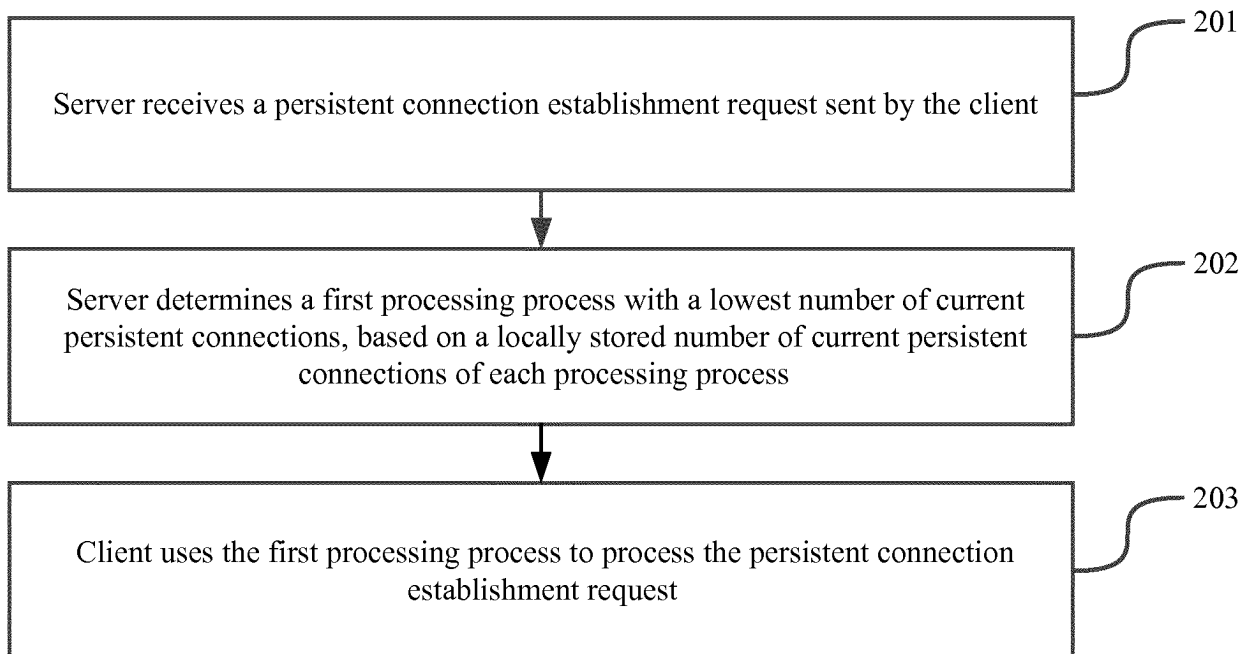


FIG. 2

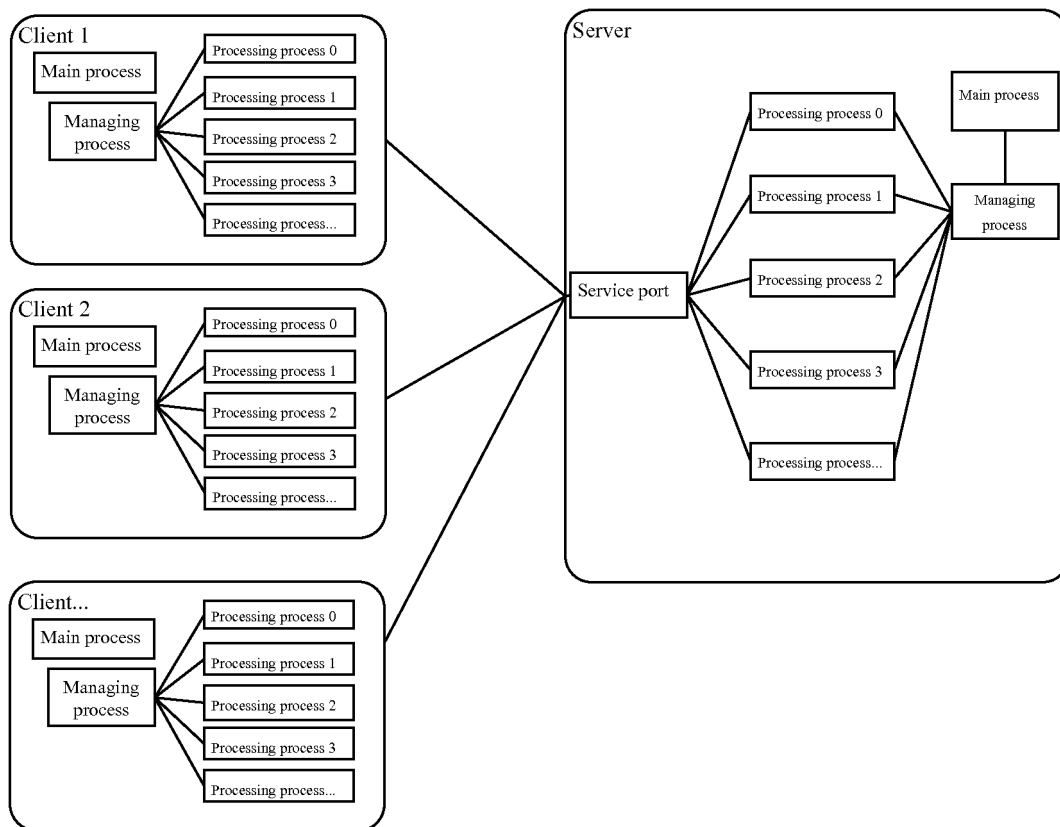


FIG. 3

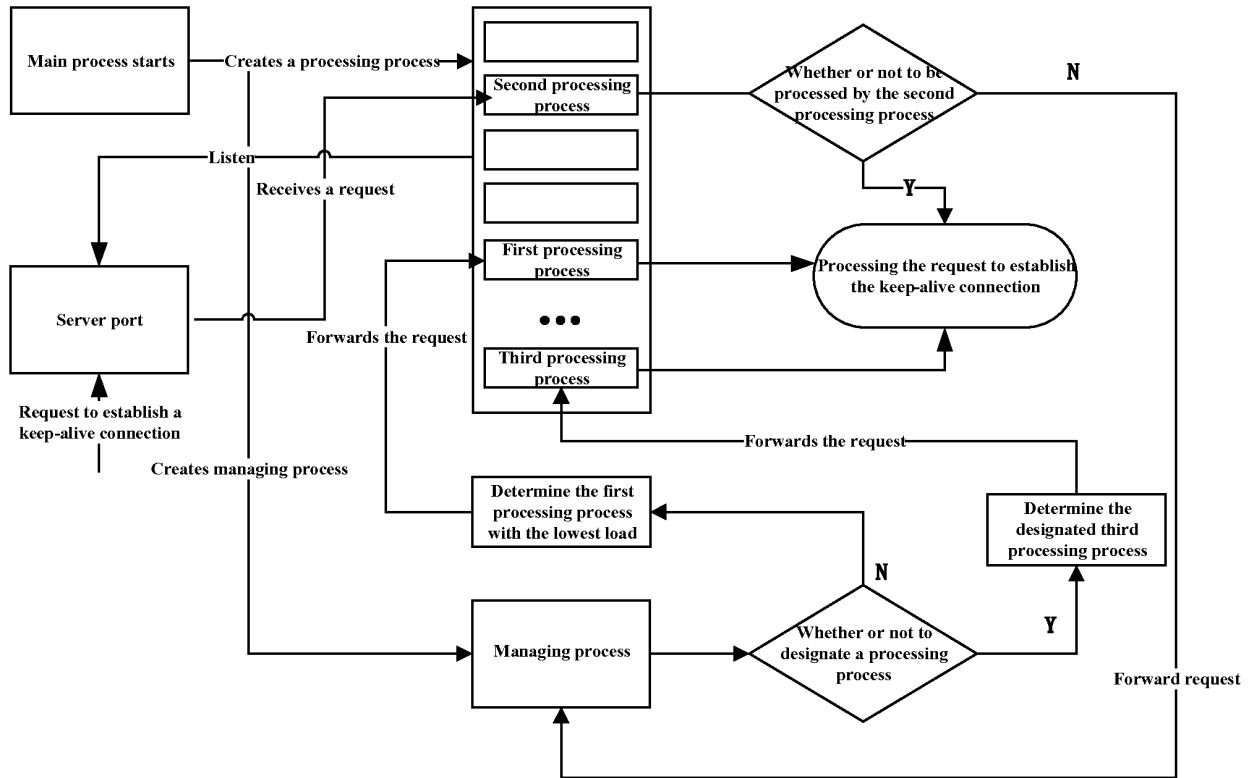


FIG. 4

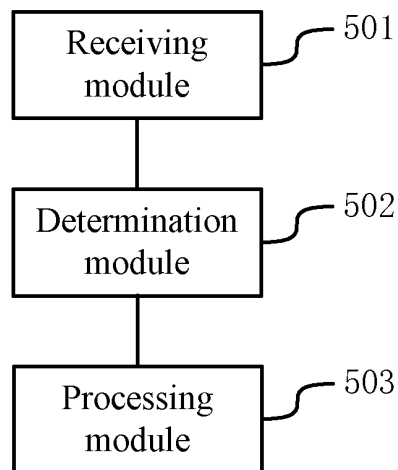


FIG. 5

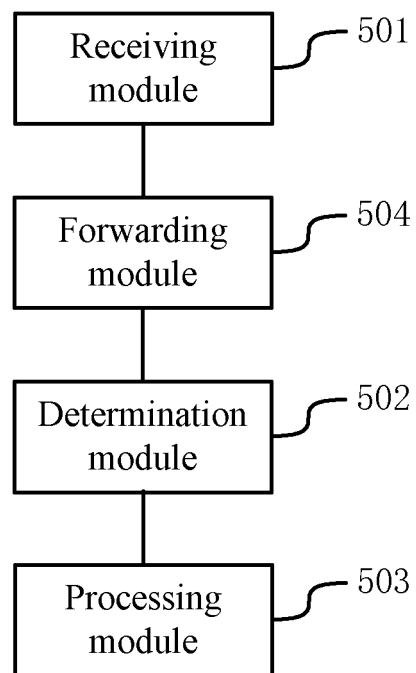


FIG. 6

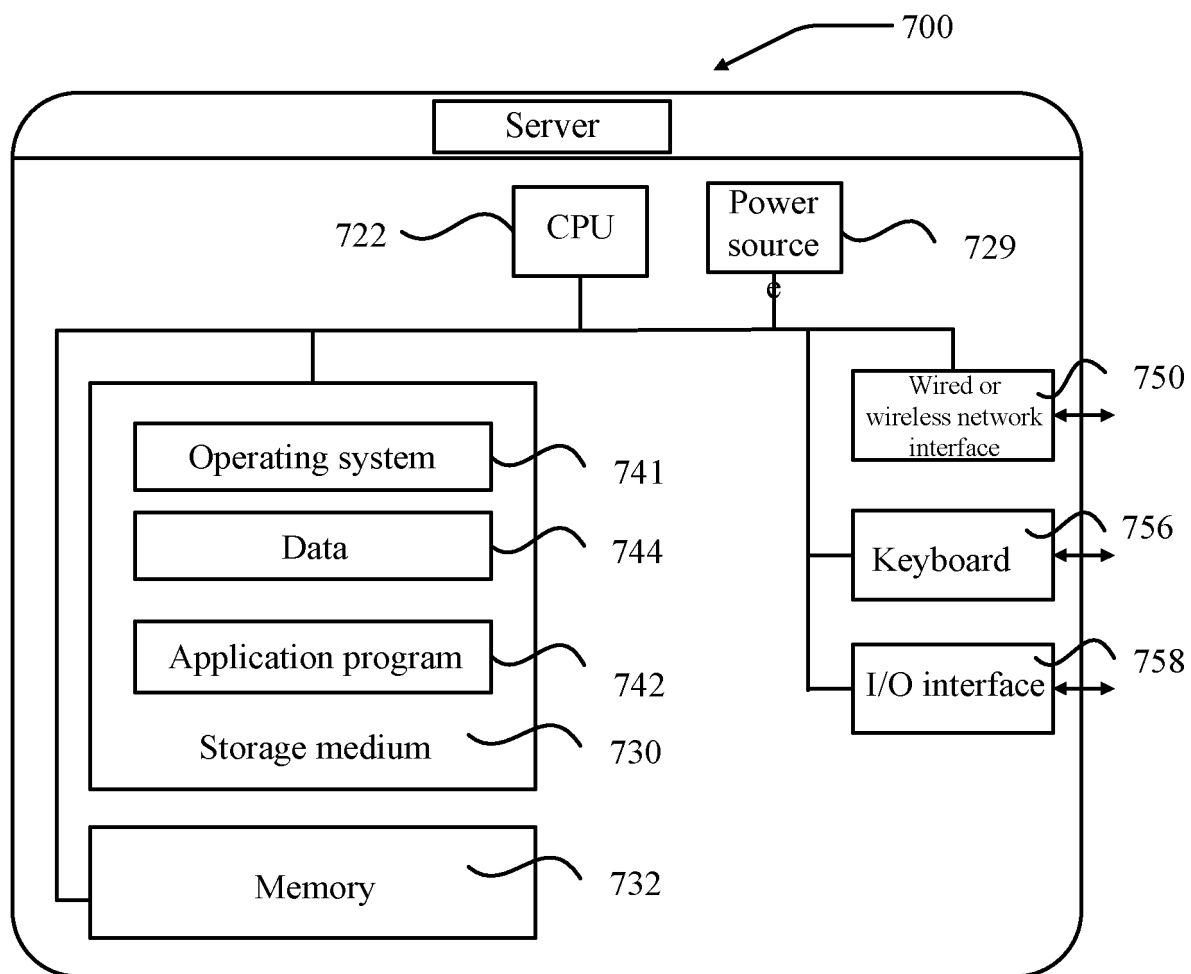


FIG. 7

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2017/095551

A. CLASSIFICATION OF SUBJECT MATTER H04L 29/08 (2006.01) i According to International Patent Classification (IPC) or to both national classification and IPC												
B. FIELDS SEARCHED												
Minimum documentation searched (classification system followed by classification symbols) H04L; H04Q; H04W												
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched												
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) CNKI, CNPAT, WPI, EPODOC: TCP, 连接, 长连接, 短连接, 请求, 消息, 进程, 最少, 最低, 阈值, 门限, 套接, connection, link, long link, short link, request, message, process, minimum, fewest, threshold, socket												
C. DOCUMENTS CONSIDERED TO BE RELEVANT												
<table border="1"> <thead> <tr> <th>Category*</th> <th>Citation of document, with indication, where appropriate, of the relevant passages</th> <th>Relevant to claim No.</th> </tr> </thead> <tbody> <tr> <td>X</td> <td>CN 102916953 A (HISENSE MEDIA NETWORKS CO., LTD.), 06 February 2013 (06.02.2013), description, paragraphs [0083]-[0170]</td> <td>1-12</td> </tr> <tr> <td>A</td> <td>CN 105721342 A (TENCENT TECHNOLOGY (SHENZHEN) CO., LTD.), 29 June 2016 (29.06.2016), entire document</td> <td>1-12</td> </tr> <tr> <td>A</td> <td>CN 105763634 A (SI-TECH INFORMATION TECHNOLOGY LTD.), 13 July 2016 (13.07.2016), entire document</td> <td>1-12</td> </tr> </tbody> </table>	Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.	X	CN 102916953 A (HISENSE MEDIA NETWORKS CO., LTD.), 06 February 2013 (06.02.2013), description, paragraphs [0083]-[0170]	1-12	A	CN 105721342 A (TENCENT TECHNOLOGY (SHENZHEN) CO., LTD.), 29 June 2016 (29.06.2016), entire document	1-12	A	CN 105763634 A (SI-TECH INFORMATION TECHNOLOGY LTD.), 13 July 2016 (13.07.2016), entire document	1-12
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.										
X	CN 102916953 A (HISENSE MEDIA NETWORKS CO., LTD.), 06 February 2013 (06.02.2013), description, paragraphs [0083]-[0170]	1-12										
A	CN 105721342 A (TENCENT TECHNOLOGY (SHENZHEN) CO., LTD.), 29 June 2016 (29.06.2016), entire document	1-12										
A	CN 105763634 A (SI-TECH INFORMATION TECHNOLOGY LTD.), 13 July 2016 (13.07.2016), entire document	1-12										
☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.												
<table border="0"> <tr> <td style="vertical-align: top;"> * Special categories of cited documents: “A” document defining the general state of the art which is not considered to be of particular relevance “E” earlier application or patent but published on or after the international filing date “L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) “O” document referring to an oral disclosure, use, exhibition or other means “P” document published prior to the international filing date but later than the priority date claimed </td> <td style="vertical-align: top;"> “T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention “X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone “Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art “&” document member of the same patent family </td> </tr> </table>	* Special categories of cited documents: “A” document defining the general state of the art which is not considered to be of particular relevance “E” earlier application or patent but published on or after the international filing date “L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) “O” document referring to an oral disclosure, use, exhibition or other means “P” document published prior to the international filing date but later than the priority date claimed	“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention “X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone “Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art “&” document member of the same patent family										
* Special categories of cited documents: “A” document defining the general state of the art which is not considered to be of particular relevance “E” earlier application or patent but published on or after the international filing date “L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) “O” document referring to an oral disclosure, use, exhibition or other means “P” document published prior to the international filing date but later than the priority date claimed	“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention “X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone “Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art “&” document member of the same patent family											
<table border="1"> <tr> <td>Date of the actual completion of the international search 01 March 2018</td> <td>Date of mailing of the international search report 23 March 2018</td> </tr> </table>	Date of the actual completion of the international search 01 March 2018	Date of mailing of the international search report 23 March 2018										
Date of the actual completion of the international search 01 March 2018	Date of mailing of the international search report 23 March 2018											
<table border="1"> <tr> <td>Name and mailing address of the ISA State Intellectual Property Office of the P. R. China No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088, China Facsimile No. (86-10) 62019451</td> <td>Authorized officer WANG, Xinyue Telephone No. (86-10) 53961795</td> </tr> </table>	Name and mailing address of the ISA State Intellectual Property Office of the P. R. China No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088, China Facsimile No. (86-10) 62019451	Authorized officer WANG, Xinyue Telephone No. (86-10) 53961795										
Name and mailing address of the ISA State Intellectual Property Office of the P. R. China No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088, China Facsimile No. (86-10) 62019451	Authorized officer WANG, Xinyue Telephone No. (86-10) 53961795											

Form PCT/ISA/210 (second sheet) (July 2009)

INTERNATIONAL SEARCH REPORT
Information on patent family membersInternational application No.
PCT/CN2017/095551

5	Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
	CN 102916953 A	06 February 2013	None	
10	CN 105721342 A	29 June 2016	None	
	CN 105763634 A	13 July 2016	None	
15				
20				
25				
30				
35				
40				
45				
50				
55				

Form PCT/ISA/210 (patent family annex) (July 2009)